ICS 35.080
L 77

中华人民共和国国家标准

GB/T 25000.10—2016
代替 GB/T 16260.1—2006

系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第10部分:系统与软件质量模型

Systems and software engineering—Systems and software Quality Requirements and Evaluation(SQuaRE)—Part 10:System and software quality models

[ISO/IEC 25010:2011,Systems and software engineering—Systems and software Quality Requirements and Evaluation(SQuaRE)—System and software quality models,MOD]

2016-10-13 发布　　2017-05-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会　发布

目　　次

前言 …… Ⅲ
引言 …… Ⅴ
1　范围 …… 1
2　规范性引用文件 …… 1
3　术语和定义 …… 2
4　质量模型框架 …… 5
4.1　结构 …… 5
4.2　使用质量模型及特性说明 …… 6
4.3　产品质量模型及特性说明 …… 8
4.4　质量模型的目标 …… 13
4.5　质量模型的使用 …… 14
4.6　来自不同利益相关方视角的质量 …… 14
4.7　各质量模型间的关系 …… 15
附录A（资料性附录）　与GB/T 16260.1—2006中质量模型的对比 …… 17
附录B（资料性附录）　与ISO/IEC 25010:2011的差异 …… 20
附录C（资料性附录）　利用质量模型测量 …… 21
参考文献 …… 26

前　　言

GB/T 25000《系统与软件工程　系统与软件质量要求和评价(SQuaRE)》分为如下部分：

——第 1 部分：SQuaRE 指南；
——第 2 部分：计划与管理；
——第 10 部分：系统与软件质量模型；
——第 12 部分：数据质量模型；
——第 20 部分：测量参考模型和指南；
——第 21 部分：质量测度元素；
——第 22 部分：使用质量测量；
——第 23 部分：系统和软件产品质量测量；
——第 24 部分：数据质量测量；
——第 30 部分：质量需求；
——第 40 部分：评价过程；
——第 41 部分：开发方、需方和独立评价方的评价指南；
——第 42 部分：评价模块；
——第 45 部分：可恢复性的评价模块；
——第 51 部分：就绪可用软件产品(RUSP)的质量要求和测试细则；
——第 60 部分：易用性测试报告行业通用格式(CIF)：易用性相关信息的通用框架；
——第 62 部分：易用性测试报告行业通用格式(CIF)
——第 63 部分：易用性的行业通用格式(CIF)：使用周境描述；
——第 64 部分：易用性的行业通用格式(CIF)：用户要求报告；
——第 65 部分：易用性的行业通用格式(CIF)：用户需求规格说明；
——第 66 部分：易用性的行业通用格式(CIF)：评价报告。

本部分为 GB/T 25000 的第 10 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分代替 GB/T 16260.1—2006《软件工程　产品质量　第 1 部分：质量模型》。与 GB/T 16260.1—2006 相比，主要技术变化如下：

——本部分中有关产品质量特性的表述由 6 个特性调整为 8 个特性，使用质量特性由 4 个特性调整为 5 个特性。相关的子特性也做了修改、调整和补充。具体的修订细节在附录 A 中列出。

本部分采用重新起草法修改采用 ISO/IEC 25010：2011《系统与软件工程　系统与软件质量要求和评价(SQuaRE)　系统与软件质量模型》。附录 B 阐明了本部分与 ISO/IEC 25010：2011 的差异。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位：上海计算机软件技术开发中心、中国电子技术标准化研究院、国家应用软件产品质量监督检验中心、广东省科技基础条件平台中心、深圳市中安测标准技术有限公司、佛山柯维光电股份有限公司、重庆市软件评测中心有限公司、南京大学、珠海南方软件网络评测中心、湖北软件评测中心、中国航天科技集团公司软件评测中心、内蒙古电子信息产品质量检验院、上海泽众软件科技有限公司、上海得元科技有限公司、上海市软件行业协会。

本部分主要起草人：蔡立志、刘振宇、冯惠、胡芸、张旸旸、曹祥琼、周悦、王威、丁晓明、李军、张子良、何志明、张毅、廖辉、薛保平、徐宝文、杨桂枝、王瑞、夏启明、黄兆森、丁为清、高海龄、巩绍飞。

本部分所代替标准的历次版本发布情况为：

——GB/T 16260—1996；

——GB/T 16260.1—2006。

引　言

软件产品和软件密集型计算机系统正越来越多地用于实现各种各样的企业和个人的功能。个人满足感、业务成功和/或人类安全的目标和宗旨的实现则依赖于高质量的软件与系统。高质量的软件产品和软件密集型计算机系统之于利益相关方是必不可少的，它可以提供价值，并避免潜在的负面影响。

软件产品和软件密集型计算机系统的利益相关方，包括开发方、需方、使用方和使用软件密集型计算机系统的企业客户。对软件和软件密集型计算机系统的质量进行全面地规范和评估是保证利益相关方利益的关键因素，可以通过定义在系统中与利益相关方的目标和宗旨相关的必要的和期望的质量特性来实现。这包括相关的软件系统和数据，以及该系统对其利益相关方的影响的质量特性。同时，尽可能使用经确认的或被广泛认可的措施和测量方法，对于规定、测量和评价质量特性至关重要。本部分中的质量模型可以用来鉴定相关的质量特性，并进一步用来建立令人满意的需求、准则和相关措施。

本部分意在结合 ISO/IEC 25000 系列国际标准的其他部分（ISO/IEC 25000 至 ISO/IEC 25099）和即将被 ISO/IEC 2504n 分部取代的 ISO/IEC 14598 一起使用。

ISO/IEC 25000 系列国际标准的组织结构如图 1 所示：

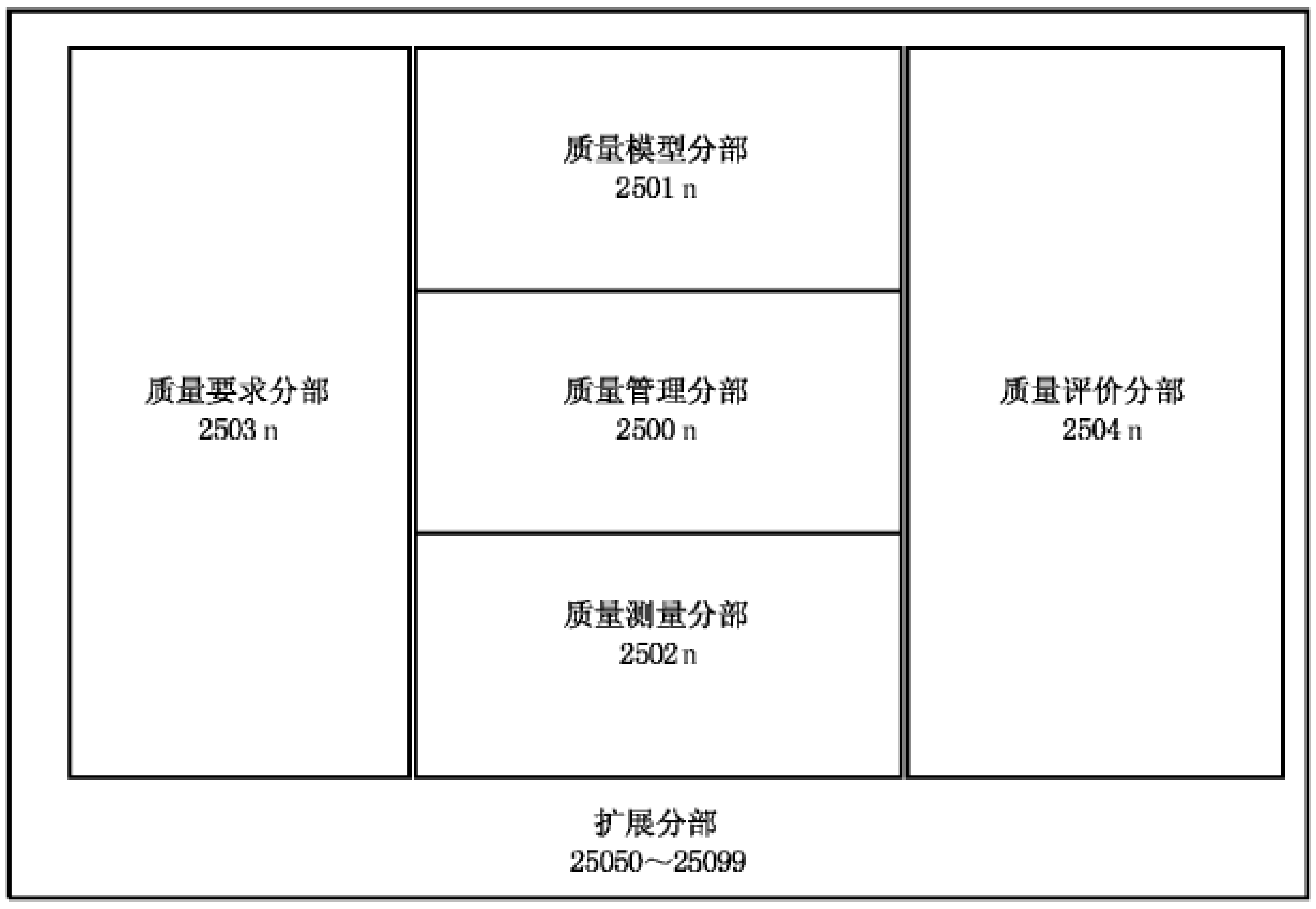


图 1　ISO/IEC 25000 系列标准组织结构

ISO/IEC 25000 系列国际标准由下列分部组成：

a)　ISO/IEC 2500n——质量管理分部。构成这个分部的标准定义了由 ISO/IEC 25000 系列标准中的所有其他标准引用的全部公共模型、术语和定义。这一分部还提供了用于负责管理软件产品质量需求和评价的支持功能的要求和指南。

b)　ISO/IEC 2501n——质量模型分部。构成这个分部的标准给出了包括计算机系统和软件产品质量、使用质量和数据的详细的质量模型。同时还提供了使用这些质量模型的实用指南。

c)　ISO/IEC 2502n——质量测量分部。构成这个分部的标准包括软件产品质量测量参考模型、质量测量的数学定义及其应用的实用指南。给出了软件内部质量、软件外部质量和使用质量测量的示例。定义并给出了构成后续测量基础的质量测度元素。

d)　ISO/IEC 2503n——质量要求分部。构成这个分部的标准有助于在质量模型和质量测量的基

础上规定质量要求。这些质量要求可用在要开发的软件产品的质量需求抽取过程中或用作评价过程的输入。

e) ISO/IEC 2504n——质量评价分部。构成这个分部的标准给出了无论由评价方、需方还是由开发方执行的软件产品评价的要求、建议和指南。还给出了作为评价模块的测量编制支持。

f) ISO/IEC 25050—25099 这是 ISO/IEC 25000 系列标准的扩展分部。目前包括了就绪可用软件的质量要求和易用性测试报告行业通用格式。

本部分的质量模型可以与 ISO/IEC 12207 和 ISO/IEC 15288 结合使用，特别是与需求定义、验证和确认相关的过程，该过程重点关注质量要求的规范和评价。ISO/IEC 25030 描述了质量模型如何在软件质量需求的过程中发挥作用，ISO/IEC 25040 描述了质量模型如何在软件质量评价的过程中发挥作用。

注 1：ISO/IEC 25030 将转化为 GB/T 25000.30。

注 2：ISO/IEC 25040 将转化为 GB/T 25000.40。

本部分可以与 ISO/IEC 15504(与软件过程评估有关)一起使用，以提供：

a) 客户-供方过程中的软件产品质量定义框架；

b) 在支持过程中对于评审、验证和确认的支持以及一个定量的质量评价框架；

c) 在管理过程中对于设置组织质量目标的支持。

本部分也可以与 GB/T 19001(与质量保证过程有关)一起使用，以提供：

a) 对于设立质量目标的支持；

b) 对于设计评审、验证和确认的支持。

系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第10部分:系统与软件质量模型

1 范围

GB/T 25000 的本部分定义了:

a) 使用质量模型,该模型由5个特性组成,每个特性又可进一步细分为一些子特性,这些特性关系到产品在特定的使用周境中使用时的交互结果。这一系统模型可以应用于整个人机系统,既包括使用中的计算机系统,也包括使用中的软件产品。

b) 产品质量模型,该模型由8个特性组成,每个特性又可进一步细分为一些子特性,这些特性关系到软件的静态性质和计算机系统的动态性质。这一模型既可以应用于计算机系统,也可以应用于软件产品。

由这两个模型所定义的特性,与所有的软件产品和计算机系统有关。这些特性和子特性为规定、测量和评价系统与软件产品质量提供了一致的术语。它们还提供了一组质量特性,依此可比较所陈述的质量需求的完备程度。

注 1:尽管产品质量模型的范围旨在软件和计算机系统,但很多特性也与更广泛的系统和服务有关。

ISO/IEC 25012:2008 中的数据质量模型是对本部分所给出的模型的补充。

注 2:ISO/IEC 25012 将转化为 GB/T 25000.12。

以上提及的两个模型,其范围不包括纯功能性性质(见 C.6),但包括功能性特性(见 4.3.2.1)。

质量模型的应用范围包括从获取、需求、开发、使用、评价、支持、维护、质量保证和审核相关的不同视角,对软件和软件密集型计算机系统的确定和评价支持。例如该模型可以被开发者、需方、质量保证与控制人员以及独立评价者,特别是那些对确定和评价软件产品质量负责的人员所使用。从质量模型的使用中能够获得好处的产品开发期间的活动,包括:

——标识系统与软件需求;

——确认详细的需求定义;

——标识系统与软件设计目标;

——标识系统与软件测试目标;

——标识作为质量保证一部分的质量控制准则;

——标识软件产品和/或软件密集型计算机系统的验收准则;

——建立支持这些活动的质量特性之测度。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅注日期的版本适用于本文件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

ISO/IEC 25012:2008 软件工程 软件产品质量要求和评价(SQuaRE) 数据质量模型[Software Engineering—Software product Quality Rquirements and Evaluation(SQuaRE)—Data quality model]

ISO/IEC 25030:2007 软件工程 软件产品质量要求和评价(SQuaRE) 质量需求[Software Engineering—Software product Quality Rquirements and Evaluation(SQuaRE)—Quality requirements]

3 术语和定义

下列术语和定义适用于本文件。

3.1

资产 asset

对个人或组织具有价值的任何东西。

注 1：改写 GB/T 19715.1—2005，定义 3.2。

注 2：在本部分中，资产是指工作产品，如需求文档、源代码模块、测度的定义等。

3.2

属性 attribute

实体的固有性质或特性，可由人工或自动化手段进行定量或定性地辨别。

注 1：GB/T 19000 区分两类属性：在事物中固有地存在的一种持久的特性；产品、过程或系统的赋予的特性（如产品价格、产品所有者）。赋予的特性并不是产品、过程或系统固有的质量特性。

注 2：改写 GB/T 25000.1—2010，定义 4.3。

3.3

基准 benchmark

可以对照其进行测量或评估结果的标准。

[ISO/IEC/IEEE 24765:2010，定义 3.256]

3.4

组件 component

在一个特定的分析层次上考虑的系统中带有分立结构的实体。诸如一个组合或软件模块。

[GB/T 18492—2001，定义 3.1]

3.5

使用周境 context of use

用户、任务、设备（硬件、软件和原材料）以及使用某产品的物理和社会环境。

[GB/T 18978.11—2004，定义 3.5]

3.6

直接用户 direct user

与产品交互的个体。

注 1：与 GB/T 18978.11—2004 中"用户"的定义相同。

注 2：在本部分中，它包括主要用户和次要用户。

3.7

最终用户 end user

最终受益于系统结果的单独个体。

注：最终用户可能是软件产品的常规操作员或是公共场合下的临时用户。

[GB/T 25000.1—2010，定义 4.14]

3.8

软件质量外部测度 external measure of software quality

包含软件在内的系统在指定条件下使用时，软件产品使得系统的行为满足明确和隐含要求的程度的测度。

注 1：在测试和运行期间，通过执行软件产品来验证和/或确认行为的属性。

示例：在测试期间发现的失效数是一种与存在于计算机系统中的故障数有关的软件质量外部测度。这两种测度不一定是相同的，因为测试不能发现所有的故障；并且在不同的情况下，某一故障会导致明显不同的失效。

注 2：基于 GB/T 25000.1—2010 中“软件外部质量”的定义。

3.9

隐含的要求　implied needs

可能未明确阐述却是实际需要的要求。

注：当软件产品用于特定场合时一些隐含的要求就成为显然的。

示例：隐含的要求包括：尽管没明确阐述但通过其他明确的要求而隐含的需要，以及由于其被认为是显然的或显而易见的而未明确阐述的需要。

[GB/T 25000.1—2010，定义 4.23]

3.10

间接用户　indirect user

接收系统的输出，但是不与系统进行交互的个人。

3.11

软件质量内部测度　internal measure of software quality

软件产品在指定条件使用时，软件产品的一组静态属性满足明确和隐含要求的程度的测度。

注 1：静态属性包括那些与软件架构、结构及其组件有关的属性。

注 2：静态属性可通过评审、审查、模拟和/或自动化工具来验证。

示例：在走查中发现的复杂的测度和故障的数量、严重程度以及失效频率，是从软件产品自身取得的软件内部质量测度。

注 3：基于 GB/T 25000.1—2010 中“软件内部质量”的定义。

3.12

测度（名）　measure（noun）

作为测量结果被赋予值的变量。

注：该术语的复数“measures”被用来指代基本测度、导出测度和指标的统称。

[ISO/IEC 15939：2007，定义 2.15]

3.13

测量（动）　measure（verb）

执行一次测量（活动）。

[GB/T 11457—2006，定义 2.922]

3.14

测量　measurement

一组操作，其目的是确定某个测度的值。

[ISO/IEC 15939：2007，定义 2.17]

注：测量可包括分配一个定性的类别，例如源程序的语言（Ada，C，COBOL 等）。

3.15

使用质量　quality in use

指定用户使用产品或系统满足其要求的程度，以达到在指定的使用周境中的有效性、效率和满意度等指定目标。

3.16

质量测度　quality measure

由一个测量函数所定义的测度，该函数具有两个或多个质量测度元素的值。

[ISO/IEC 25021 定义 4.14]

3.17

质量测度元素　quality measure element

为了量化一个属性而通过该属性和测量方法所定义的测度，包括有选择地通过一个数学函数的转换。

[ISO/IEC 25021 定义 4.15]

3.18

质量模型　quality model

定义的特征集以及它们之间的关系集，为规约质量需求以及评价质量提供了一个框架。

[GB/T 25000.1—2010，定义 4.44]

3.19

质量属性　quality attribute

可测量的质量要素(组成部分)。

3.20

风险　risk

一个给定威胁发生的概率及该威胁发生后的潜在不利后果的函数。

[GB/T 18492—2001，定义 3.12]

3.21

软件产品　software product

一组计算机程序、规程以及可能的相关文档和数据。

注 1： 产品包括中间产品和意图用于开发方和维护方使用的产品。

注 2： 在 ISO/IEC 25000 系列标准中，软件质量和软件产品质量具有同样的含义。

[GB/T 25000.1—2010，定义 4.49]

3.22

软件质量　software quality

在规定条件下使用时，软件产品满足明确和隐含要求的能力。

注： 该定义不同于 GB/T 19000—2008 中的质量定义，在本部分中软件质量是指满足明确和隐含要求，而在 GB/T 19000 质量定义是指满足要求。

[GB/T 25000.1—2010，定义 4.51]

3.23

软件质量特性　software quality characteristic

支撑软件质量的软件质量属性的类别。

注： 软件质量特性可细化为多层的子特性，最终细化为软件质量属性。

[GB/T 25000.1—2010，定义 4.52]

3.24

软件质量需求　software quality requirement

对软件中现有的质量属性的需求。

3.25

利益相关方　stakeholder

对系统或系统特性有权利、共享、主张或兴趣以满足其需要和期望的个体或组织。

3.26

系统　system

为达到一个或多个明确目的而组织起来的、相互作用的元素的组合体。

注 1： 一个系统可被认为是一个产品或它提供的服务。

注 2： 实际上，对系统含义的解释通常通过使用一个联合名词来阐明，如飞行器系统。或者，单词“系统”可简单地由上下文相关的同义词来替代，如飞行器，虽然这可能使系统的观点不太明显。

[GB/T 22032—2008，定义 4.17]

3.27

用户　user

在系统使用过程中，与系统进行交互或从系统中获益的个人或组织。

注：主要用户和次要用户与系统进行交互，且主要用户和间接用户可以从系统中获益（见 4.6）。

3.28

确认　validation

通过检查和提供客观证据来证实针对某一特定预期用途的需求已经得到满足。

注 1：“确认过的”一词用来表示相应的状况。

注 2：在设计和开发中，确认涉及检查某个产品以确定是否符合用户需要的过程。

注 3：确认通常是对最终产品在规定的使用条件下进行的。在早期阶段，也可能需要进行确认。

注 4：如果有不同的预期用途，可以进行多重确认。

[GB/T 25000.1—2010，定义 4.62]

3.29

验证　verification

通过检查和提供客观的证据来证实规定需求已经得到满足。

注 1：“验证过的”用来表示相应的状况。

注 2：在设计和开发中，验证是指对某项规定活动的结果进行检查的过程，以确定该活动对规定需求的符合情况。

[GB/T 25000.1—2010，定义 4.64]

4　质量模型框架

4.1　结构

系统的质量是指该系统满足其不同利益相关方所明确的或隐含的要求的程度，并由此提供了相应的价值。这些明确的和隐含的要求是通过 ISO/IEC 25000 系列标准中的质量模型予以表达的，其中把产品质量分类为一些特性，并在某些情况中进一步被分解为子特性（某些子特性又被分解为子子特性）。这一层次化的分解，提供了便利的产品质量分级。但是，本部分选择与某个特性关联的子特性集作为典型关注的代表，而没有穷取其所有子特性。

一个系统可测量的、与质量相关的属性被称为与质量测度相关联的质量属性。达到该质量特性或子特性的测度，除非可直接测量该特性或子特性，否则，就有必要标识那些覆盖该特性或子特性的一组属性，获得每一个质量测度，并通过计算把它们组合起来，以达到对应质量特性或子特性的一个导出的质量测度（参见附录 C）。图 2 给出了质量特性、子特性和质量属性之间的关系。

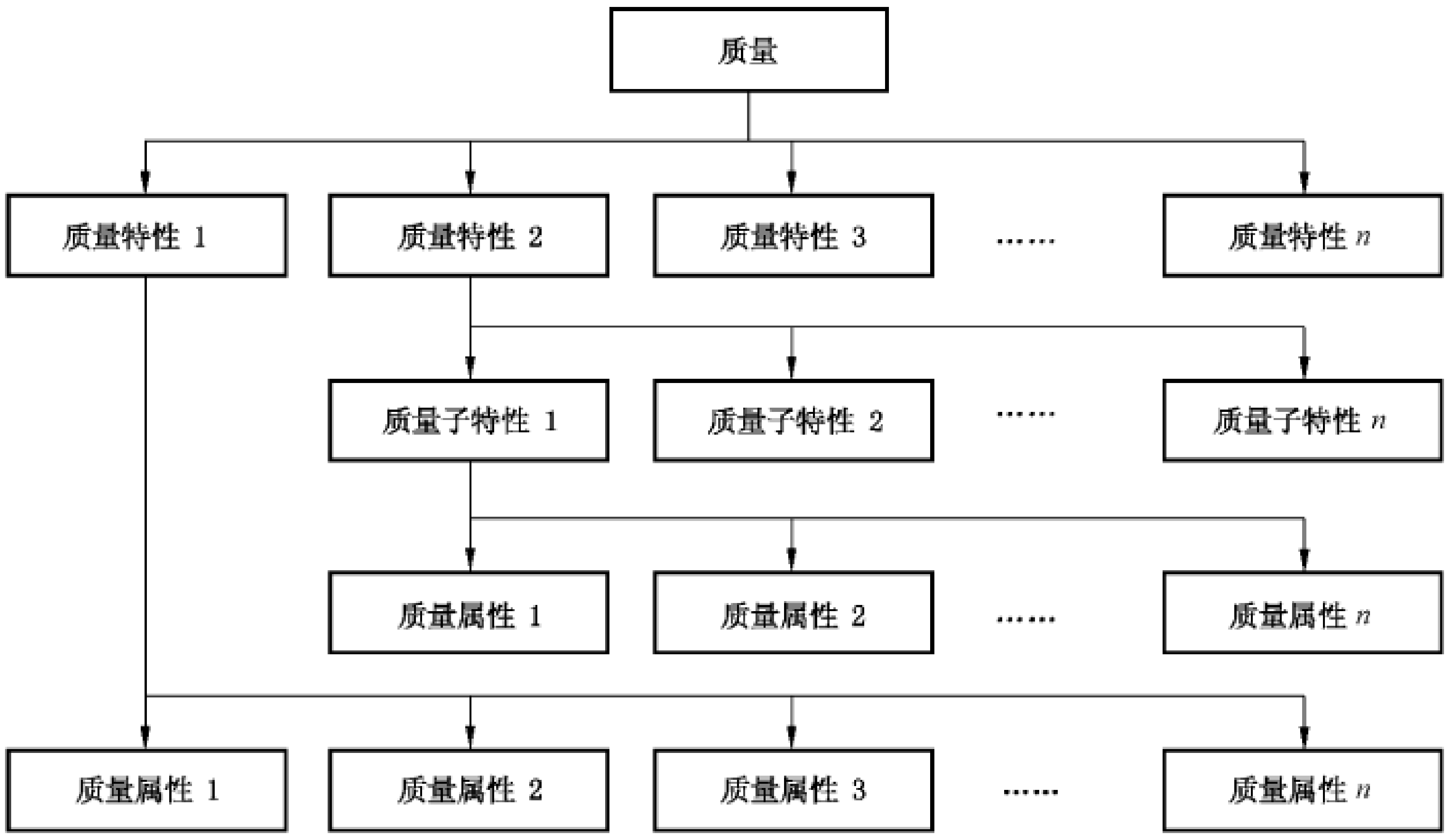


图 2　质量模型结构图

当前，ISO/IEC 25000 系列标准提供了 3 个质量模型，包括本部分中的使用质量模型和产品质量模型，以及 ISO/IEC 25012:2008 中的数据质量模型。这些质量模型合起来作为一个框架，以确保所有质量特性得到考虑。这些模型为相关的广泛利益相关方，如软件开发方、系统集成方、需方、所有者、维护者、合同方、质量保证和控制人员、用户，提供了一个质量特性集。

贯穿这些模型的质量特性全集，并非相关于每一个利益相关方。但是，每类利益相关方在确定要使用的质量特性集之前，宜参与评审和考虑每个模型中相关的质量特性，例如，确立产品和系统的性能需求或评价准则。

4.2 使用质量模型及特性说明

4.2.1 使用质量模型

使用质量模型将使用质量属性划分为 5 个特性：有效性、效率、满意度、抗风险和周境覆盖（见图 3）。每个特性都可以被赋予到利益相关方的不同的活动中，例如操作人员的交互或开发人员的维护。

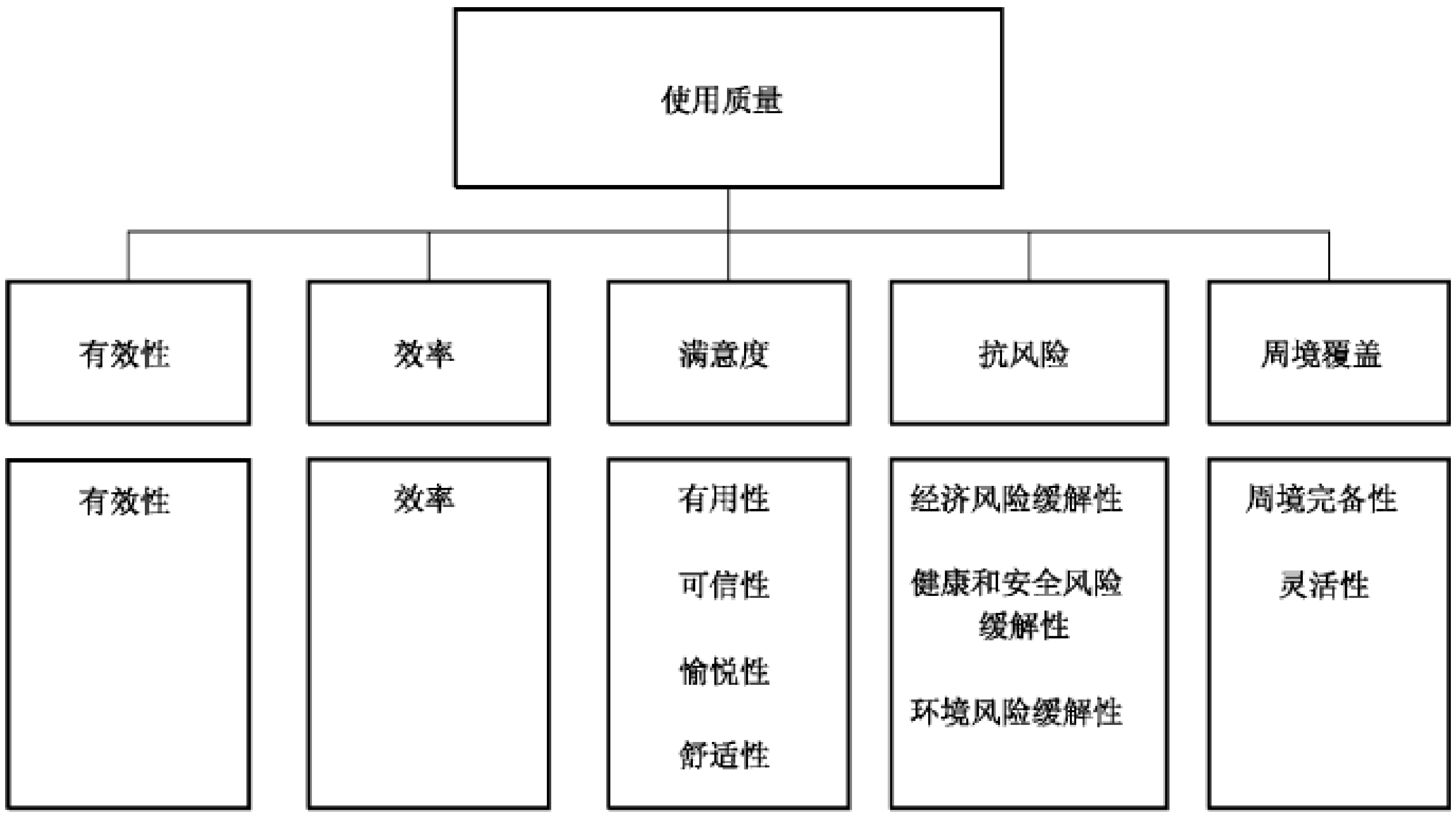


图 3 使用质量模型

系统的使用质量描述了产品（系统或软件产品）对利益相关方造成的影响。它是由软件、硬件和运行环境的质量，以及用户、任务和社会环境的特性所决定的。所有这些因素均有利于系统的使用质量。

4.2.2 针对使用质量模型中的每个质量特性给出了说明。

GB/T 16260.4（将被 GB/T 25000.22 取代）给出了使用质量测度的实例。

注：ISO/IEC 25022 将转化为 GB/T 25000.22。

4.2.2 特性说明

4.2.2.1 有效性

用户实现指定目标的准确性和完备性。

4.2.2.2 效率

与用户实现目标的准确性和完备性相关的资源消耗。

注：相关的资源可包括完成任务的时间（人力资源）、原材料或使用的财务成本。

4.2.2.3 满意度

产品或系统在指定的使用周境中使用时，用户的要求被满足的程度。

注 1：对于不直接与产品或系统交互的用户，仅与目标实现和可信性相关。

注 2：满意度是指用户对其与产品或系统交互的反应，包括对产品使用的态度。

4.2.2.3.1 有用性

用户对实用目标的实现感到满意的程度，包括使用的结果和使用后产生的后果。

4.2.2.3.2 可信性

用户或者其他利益相关方对产品或系统将如预期地运行有信心的程度。

4.2.2.3.3 愉悦度

用户因个人要求被满足而获得愉悦感的程度。

注：个人要求可包括获得新的知识和技能、进行个性化交流和引发愉快的回忆。

4.2.2.3.4 舒适性

用户生理上感到舒适的程度。

4.2.2.4 抗风险

产品或系统在经济现状、人的生命、健康或环境方面缓解潜在风险的程度。

4.2.2.4.1 经济风险缓解性

在预期的使用周境中，产品或系统在经济现状、高效运行、商业财产、信誉或其他资源方面缓解潜在风险的程度。

4.2.2.4.2 健康和安全风险缓解性

在预期的使用周境中，产品或系统缓解人员潜在风险的程度。

4.2.2.4.3 环境风险缓解性

在预期的使用周境中，产品或系统在财产或环境方面缓解潜在风险程度。

4.2.2.5 周境覆盖

在指定的使用周境和超出最初设定需求的周境中，产品或系统在有效性、效率、抗风险和满意度特性方面能够被使用的程度。

注：使用周境与使用质量和一些产品质量特性或子特性相关(即“指定条件”)。

4.2.2.5.1 周境完备性

在所有指定的使用周境中，产品或系统在有效性、效率、抗风险和满意度特性方面能够被使用的程度。

注：周境完备性既可以当作在所有预期的使用周境中产品在有效性、效率、抗风险和满意度特性方面为达到指定目标被指定用户使用的程度，也可以通过在所有预期的使用周境中支持使用的产品属性的存在来进行指定或测量。

示例：在小屏幕、低带宽、非专业人员操作以及软件的容错模式(例如没有网络连接)的条件下软件的可用程度。

4.2.2.5.2 灵活性

在超出最初设定需求的周境中，产品或系统在有效性、效率、抗风险和满意度特性方面能够被使用的程度。

注 1：灵活性可通过使产品适应于额外的用户组、任务和文化来获得(见 4.3.2.8.1)。

注 2：灵活性使产品考虑现状、机会和个人喜好等非预期因素。

注 3：如果产品设计时未考虑灵活性，那么在预期之外的周境下使用该产品可能是不安全的。

注 4：灵活性既可以当作在附加类型的使用周境中产品在有效性、效率、抗风险和满意度特性方面为达到附加类型的目标被附加类型的用户使用的程度，也可以通过修改以适应新型用户、任务和环境的能力，以及 ISO 9241-110 中定义的个性化的适宜性来进行测量。

4.3 产品质量模型及特性说明

4.3.1 产品质量模型

产品质量模型将系统/软件产品质量属性划分为 8 个特性：功能性、性能效率、兼容性、易用性、可靠性、信息安全性、维护性和可移植性。每个特性由一组相关子特性组成(见图 4)。

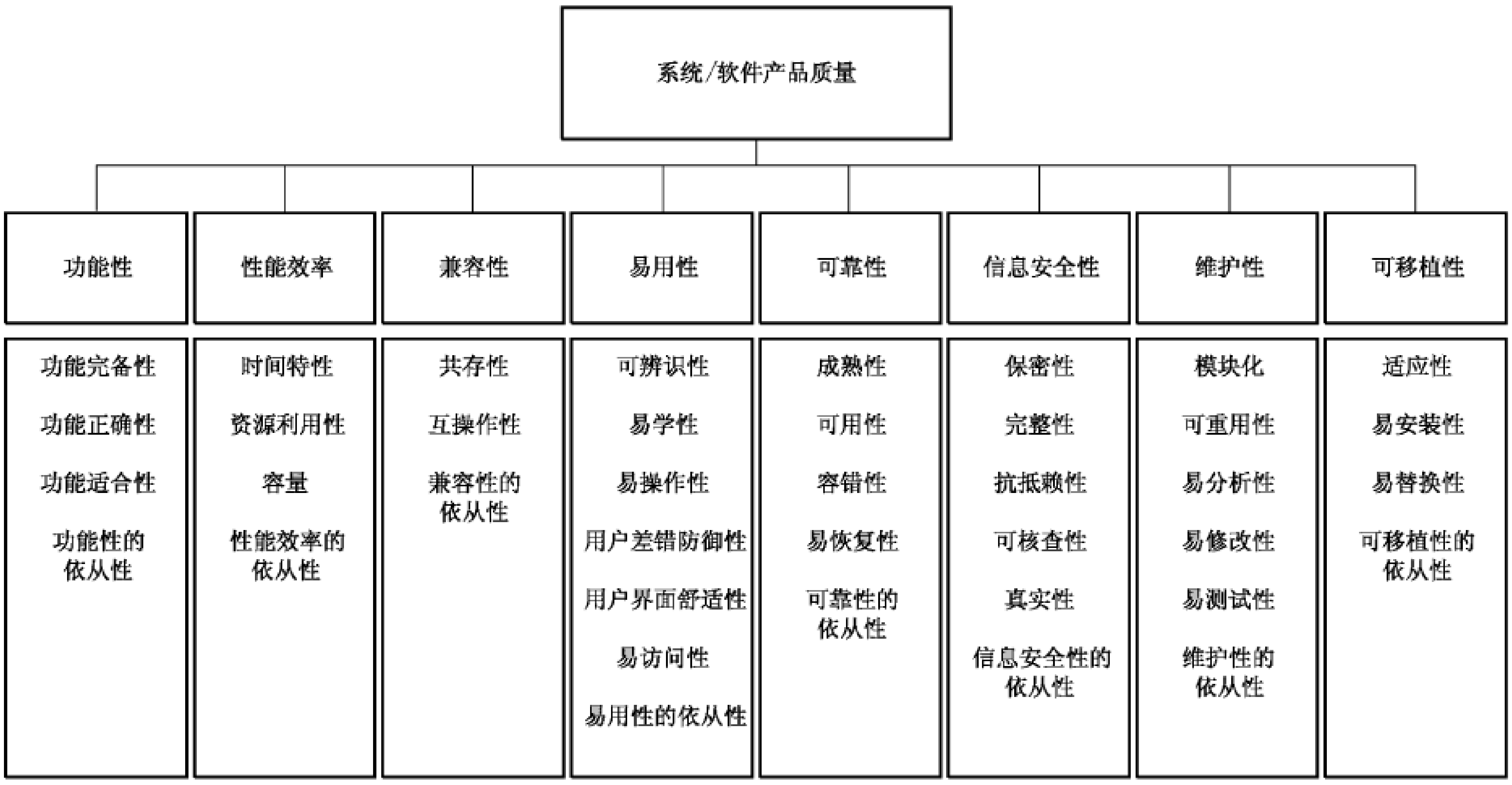


图 4　产品质量模型

产品质量模型可以只应用于软件产品，或者包含软件的计算机系统，因为大多数子特性与软件和系统相关。

4.3.2 针对产品质量模型中的每个质量特性给出了说明。

4.3.2 特性说明

4.3.2.1 功能性

在指定条件下使用时，产品或系统提供满足明确和隐含要求的功能的程度。

注：功能性只关注功能是否满足明确和隐含要求，而不是功能规格说明(见 C.6)。

4.3.2.1.1 功能完备性

功能集对指定的任务和用户目标的覆盖程度。

4.3.2.1.2 功能正确性

产品或系统提供具有所需精度的正确的结果的程度。

4.3.2.1.3 功能适合性

功能促使指定的任务和目标实现的程度。

示例:不含任何不必要的步骤,只提供用户必要的步骤就可以完成任务。

注:功能适合性相当于 ISO 9241-110 中的任务的适合性。

4.3.2.1.4 功能性的依从性

产品或系统遵循与功能性相关的标准、约定或法规以及类似规定的程度。

4.3.2.2 性能效率

性能与在指定条件下所使用的资源量有关。

注:资源可包括其他软件产品、系统的软件和硬件配置、以及原材料(如打印纸和存储介质)。

4.3.2.2.1 时间特性

产品或系统执行其功能时,其响应时间、处理时间及吞吐率满足需求的程度。

4.3.2.2.2 资源利用性

产品或系统执行其功能时,所使用资源数量和类型满足需求的程度。

注:人力资源属于效率(见 4.2.2.2)的一部分。

4.3.2.2.3 容量

产品或系统参数的最大限量满足需求的程度。

注:参数可包括存储数据项数量、并发用户数、通信带宽、交易吞吐量和数据库规模。

4.3.2.2.4 性能效率的依从性

产品或系统遵循与性能效率相关的标准、约定或法规以及类似规定的程度。

4.3.2.3 兼容性

在共享相同的硬件或软件环境的条件下,产品、系统或组件能够与其他产品、系统或组件交换信息,和/或执行其所需的功能的程度。

4.3.2.3.1 共存性

在与其他产品共享通用的环境和资源的条件下,产品能够有效执行其所需的功能并且不会对其他产品造成负面影响的程度。

4.3.2.3.2 互操作性

两个或多个系统、产品或组件能够交换信息并使用已交换的信息的程度。

4.3.2.3.3　兼容性的依从性

产品或系统遵循与兼容性相关的标准、约定或法规以及类似规定的程度。

4.3.2.4　易用性

在指定的使用周境中,产品或系统在有效性、效率和满意度特性方面为了指定的目标可为指定用户使用的程度。

注:易用性既可以从它的子特性角度当作产品质量特性来进行指定或测量,也可以直接通过测度(使用质量的子集)来进行指定或测量。

4.3.2.4.1　可辨识性

用户能够辨识产品或系统是否适合他们的要求的程度(参见 4.3.2.1.3 功能适合性)。

注 1:可辨识性将取决于通过对产品或系统的初步印象和/或任何相关文档来辨识产品或系统功能的能力。

注 2:产品或系统提供的信息可包括演示、教程、文档或网站的主页信息。

4.3.2.4.2　易学性

在指定的使用周境中,产品或系统在有效性、效率、抗风险和满意度特性方面为了学习使用该产品或系统这一指定的目标可为指定用户使用的程度。

注:易学性既可以被当作在指定的使用周境中产品或系统在有效性、效率、抗风险和满意度特性方面为了学习使用该产品或系统这一指定的目标被指定用户使用的程度,也可以通过相当于 ISO 9241-110 中定义的学习的适宜性的产品属性来进行指定或测量。

4.3.2.4.3　易操作性

产品或系统具有易于操作和控制的属性的程度。

注:易操作性相当于 ISO 9241-110 中定义的可控性、(操作)容错性和与用户期望的符合性。

4.3.2.4.4　用户差错防御性

系统预防用户犯错的程度。

4.3.2.4.5　用户界面舒适性

用户界面提供令人愉悦和满意的交互的程度。

注:这涉及产品或系统旨在提高用户愉悦性和满意度的各种属性,诸如颜色的使用和图形化设计的自然性。

4.3.2.4.6　易访问性

在指定的使用周境中,为了达到指定的目标,产品或系统被具有最广泛的特征和能力的个体所使用的程度。

注 1:能力的范围包括与年龄有关的能力障碍。

注 2:对具有能力障碍的人而言,易访问性既可以被当作在指定的使用周境中产品或系统在有效性、效率、抗风险和满意度特性方面为了指定的目标被具有指定能力障碍的用户使用的程度,也可以通过支持易访问性的产品属性来进行指定或测量。

4.3.2.4.7　易用性的依从性

产品或系统遵循与易用性相关的标准、约定或法规以及类似规定的程度。

4.3.2.5 可靠性

系统、产品或组件在指定条件下、指定时间内执行指定功能的程度。

注 1：软件中不会发生损耗。可靠性的种种局限是由需求、设计和实现中的故障或周境的变化所致。

注 2：IEC 60050-191 中的可靠性(dependability)特性包括可用性和它的固有的或外部的影响因素，诸如有用性、可靠性(包括容错性和易恢复性)、信息安全性(包括保密性和完整性)、维护性、耐用性和维护支持。

4.3.2.5.1 成熟性

系统、产品或组件在正常运行时满足可靠性要求的程度。

注：成熟性这个概念可以被用于其他质量特性中，以表明它们在正常运行时满足需求的程度。

4.3.2.5.2 可用性

系统、产品或组件在需要使用时能够进行操作和访问的程度。

注：可用性可以通过系统、产品或组件在总时间中处于可用状态的百分比进行外部评估。因此，可用性是成熟性(控制失效的频率)、容错性和易恢复性(控制每个失效发生后的宕机时间长短)的组合。

4.3.2.5.3 容错性

尽管存在硬件或软件故障，系统、产品或组件的运行符合预期的程度。

4.3.2.5.4 易恢复性

在发生中断或失效时，产品或系统能够恢复直接受影响的数据并重建期望的系统状态的程度。

注：在失效发生后，计算机系统有时会宕机一段时间，这段时间的长短由其易恢复性决定。

4.3.2.5.5 可靠性的依从性

产品或系统遵循与可靠性相关的标准、约定或法规以及类似规定的程度。

4.3.2.6 信息安全性

产品或系统保护信息和数据的程度，以使用户、其他产品或系统具有与其授权类型和授权级别一致的数据访问度。

注 1：信息安全性不仅适用于存储在产品或系统中的数据或者通过产品或系统存储的数据，也适用于传输中的数据。

注 2：存活性(在受到攻击时，产品或系统及时提供必要的服务，继续履行其任务的程度)包含在“易恢复性(见 4.3.2.5.4)”中。

注 3：免疫性(产品或系统抗攻击的程度)包含在完整性(见 4.3.2.6.2)中。

注 4：信息安全性有利于可信性(见 4.2.2.3.2)。

4.3.2.6.1 保密性

产品或系统确保数据只有在被授权时才能被访问的程度。

4.3.2.6.2 完整性

系统、产品或组件防止未授权访问、篡改计算机程序或数据的程度。

4.3.2.6.3 抗抵赖性

活动或事件发生后可以被证实且不可被否认的程度。

4.3.2.6.4 可核查性

实体的活动可以被唯一地追溯到该实体的程度。

4.3.2.6.5 真实性

对象或资源的身份标识能够被证实符合其声明的程度。

4.3.2.6.6 信息安全性的依从性

产品或系统遵循与信息安全性相关的标准、约定或法规以及类似规定的程度。

4.3.2.7 维护性

产品或系统能够被预期的维护人员修改的有效性和效率的程度。

注 1： 修改包括纠正、改进或软件对环境、需求和功能规格说明变化的适应。修改包括那些由专业支持人员实施的，以及那些由业务或操作人员、最终用户实施的。

注 2： 维护性包括安装更新和安装升级。

注 3： 维护性可以被解释为便于维护活动的一种产品或系统固有能力，或者为了产品或系统维护的目标维护人员所经历的使用质量。

4.3.2.7.1 模块化

由多个独立组件组成的系统或计算机程序，其中一个组件的变更对其他组件的影响最小的程度。

4.3.2.7.2 可重用性

资产能够被用于多个系统，或其他资产建设的程度。

4.3.2.7.3 易分析性

可以评估预期变更(变更产品或系统的一个或多个部分)对产品或系统的影响、诊断产品的缺陷或失效原因、识别待修改部分的有效性和效率的程度。

注： 实现包括为产品或系统提供机制，以分析其自身故障以及在失效或其他事件前提供报告。

4.3.2.7.4 易修改性

产品或系统可以被有效地、有效率地修改，且不会引入缺陷或降低现有产品质量的程度。

注 1： 实现包括编码、设计、文档和验证的变更。

注 2： 模块化(见 4.3.2.7.1)和易分析性(见 4.3.2.7.3)会影响到易修改性。

注 3： 易修改性是易改变性和稳定性的组合。

4.3.2.7.5 易测试性

能够为系统、产品或组件建立测试准则，并通过测试执行来确定测试准则是否被满足的有效性和效率的程度。

4.3.2.7.6 维护性的依从性

产品或系统遵循与维护性相关的标准、约定或法规以及类似规定的程度。

4.3.2.8 可移植性

系统、产品或组件能够从一种硬件、软件、或者其他运行(或使用)环境迁移到另一种环境的有效性

和效率的程度。

注：可移植性可以被理解成产品或系统促进移植活动的一种固有能力，或移植人员为了产品或系统移植的目标体验到的使用质量。

4.3.2.8.1 适应性

产品或系统能够有效地、有效率地适应不同的或演变的硬件、软件、或者其他运行（或使用）环境的程度。

注 1：适应性包括内部能力（例如屏幕域、表、事务量、报告格式等）的可伸缩性。

注 2：适应包括那些由专业支持人员实施的，以及那些由业务或操作人员，或最终用户实施的。

注 3：如果系统能被最终用户所适应，那么适应性就相当于 ISO 9241-110 中所定义的个性化的适宜性。

4.3.2.8.2 易安装性

在指定环境中，产品或系统能够成功地安装和/或卸载的有效性和效率的程度。

注：如果系统或产品能被最终用户所安装，那么易安装性会影响到所产生的功能合适性和易操作性。

4.3.2.8.3 易替换性

在相同的环境中，产品能够替换另一个相同用途的指定软件产品的程度。

注 1：软件产品的新版本的易替换性在升级时对于用户来说是重要的。

注 2：易替换性可包括易安装性和适应性的属性。鉴于其重要性，易替换性作为一个独立的子特性被引入。

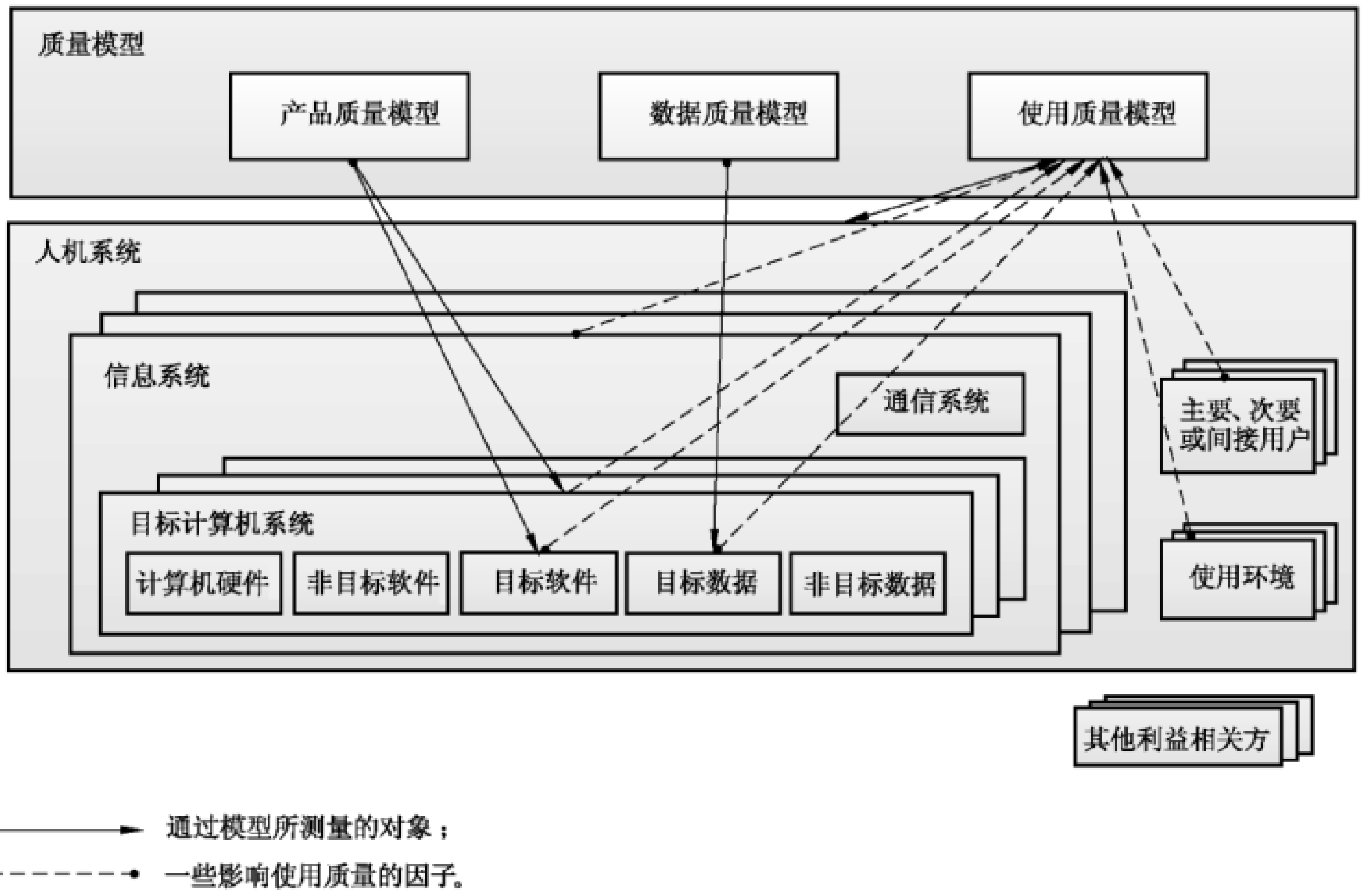
注 3：易替换性将降低锁定风险；因此其他软件产品可以代替当前产品，例如按标准文档格式使用。

4.3.2.8.4 可移植性的依从性

产品或系统遵循与可移植性相关的标准、约定或法规以及类似规定的程度。

4.4 质量模型的目标

图 5 刻画了质量模型及其相关实体的目标。



说明：

通过模型所测量的对象；

一些影响使用质量的因子。

注：此图与 ISO/IEC 25012 中的图 2 和 ISO/IEC 25030 中的图 5 在概念上相同，但却是一个关注质量模型的不同版本。

图 5 质量模型的目标

产品质量模型关注包含了目标软件产品的目标计算机系统;而使用质量模型关注包含了目标计算机系统和目标软件产品的人机系统。目标计算机系统中还包含了计算机硬件、非目标软件产品、非目标数据以及目标数据,即数据模型的主体(见C.8)。目标计算机系统包含在信息系统中,该信息系统还可能包含一个或多个计算机系统和通信系统,如局域网和广域网。该信息系统是范围更广的人机系统(如企业系统、嵌入式系统或大型控制系统)的一部分,并可能包含用户和技术与物理使用环境。其中,系统边界的判定取决于需求或评价的范围,以及用户是谁。

示例:在一个基于计算机的飞行控制系统中,如果将乘客当作用户,那么他们依赖的系统就包含机组人员、飞机机架(机体)以及飞行控制系统中的硬件和软件;而如果将机组人员当作用户,那么他们依赖的系统只由机体和飞行控制系统组成。

其他利益相关方,诸如软件开发方、系统集成方、需方、所有者、维护者、合同方、质量保证和控制人员,也会关注质量。

4.5 质量模型的使用

产品质量模型和使用质量模型,对确定质量需求、建立质量测度以及执行质量评价均是有用的(参见附录C)。本部分定义的诸质量特性可作为一个检查表予以使用,以便确保综合处理质量需求,由此为估算以后系统开发期间所需要的工作和活动奠定基础。在确定或评价计算机系统或软件产品质量时,可以将产品质量模型和使用质量模型中的特性,作为一个集合予以使用。

对一个大型计算机系统或软件产品而言,确定或测量所有部分的所有子特性是不切实际的。类似地,不可能针对所有可能的用户任务场景确定或测量使用质量。质量特性相对重要性取决于项目的高层目标和宗旨。因此,宜在将模型作为需求分解的一部分使用之前,依据利益相关方的产品目标和宗旨,对模型进行剪裁,以便标识最重要的、作为在不同类型测量之间分配的资源的那些特性和子特性。

4.6 来自不同利益相关方视角的质量

质量模型为汇总各利益相关方的要求提供了一个框架。利益相关方包括以下类型的用户:

a) 主要用户:为了达到主要目标与系统进行交互的人。

b) 次要用户:提供支持的人,例如:

   1) 内容提供者、系统管理者、安全管理者;

   2) 维护人员、分析人员、移植人员、安装人员。

c) 间接用户:接收系统的输出,但不与系统进行交互的人。

在特定的使用周境中,每类用户均对使用质量和产品质量有一定的要求,表1以问句的形式给出了用户和质量特性的一些示例。

注:内容提供者对数据质量也有一定要求。

表1中的用户要求提供了需求出发点的示例,并可作为一个基础,用于测量该系统的质量对使用和维护的影响。

在软件开发或获取之前,宜从利益相关方的视角定义质量需求。分析使用需求,将产生一个产品达到使用需求所需要的导出功能和质量需求。

示例:对系统可靠性的整体要求,可导出软件产品成熟性、可用性、容错性以及可恢复性的特定需求。可靠性对整体系统的有效性、效率、抗风险和满意度也具有一定的影响。

表 1　使用质量和产品质量的用户要求示例

用户要求	主要用户	次要用户		间接用户
		内容提供者	维护人员	
	交互	交互	维护或移植	使用输出
有效性	用户使用该系统执行任务时，需要何种有效程度？	内容提供者更新该系统时，需要何种有效程度？	维护人员维护或移植该系统时，需要何种有效程度？	用户使用该系统输出时，需要何种有效程度？
效率	用户使用该系统执行任务时，需要何种效率程度？	内容提供者更新该系统时，需要何种效率程度？	维护人员维护或移植该系统时，需要何种效率程度？	用户使用该系统输出时，需要何种效率程度？
满意度	用户使用该系统执行任务时，需要何种满意程度？	内容提供者更新该系统时，需要何种满意程度？	维护人员维护或移植该系统时，需要何种满意程度？	用户使用该系统输出时，需要何种满意程度？
抗风险	用户使用该系统执行任务时，需要何种抗风险程度？	内容提供者更新该系统时，需要何种抗风险程度？	维护人员维护或移植该系统时，需要何种抗风险程度？	用户使用该系统输出时，需要何种抗风险程度？
可靠性	用户使用该系统执行任务时，需要何种可靠程度？	内容提供者更新该系统时，需要何种可靠程度？	维护人员维护或移植该系统时，需要何种可靠程度？	用户使用该系统输出时，需要何种可靠程度？
信息安全性	用户使用该系统执行任务时，需要何种信息安全程度？	内容提供者更新该系统时，需要何种信息安全程度？	维护人员维护或移植该系统时，需要何种信息安全程度？	用户使用该系统输出时，需要何种信息安全程度？
周境覆盖	用户使用该系统执行任务时，需要何种周境覆盖程度？	内容提供者更新该系统时，需要何种周境覆盖程度？	维护人员维护或移植该系统时，需要何种周境覆盖程度？	用户使用该系统输出时，需要何种周境覆盖程度？
易学性	用户使用该系统执行任务时，需要何种易学程度？	内容提供者更新该系统时，需要何种易学程度？	维护人员维护或移植该系统时，需要何种易学程度？	用户使用来自该系统输出时，需要何种易学程度？
易访问性	用户使用该系统执行任务时，需要何种访问程度？	内容提供者更新该系统时，需要何种访问程度？	维护人员维护或移植该系统时，需要何种访问程度？	用户使用来自该系统输出时，需要何种访问程度？

4.7　各质量模型间的关系

软件产品和计算机系统的属性决定了特定使用周境中的产品质量(见表 2)。

对主要用户而言，功能性、性能效率、易用性、可靠性、信息安全性对使用质量具有重大影响。其中，性能效率、可靠性和信息安全性还受到专业从事这些领域的利益相关方的关注。

对维护系统的次要用户而言，兼容性、维护性和可移植性将对使用质量具有重大影响。

表 2　质量特性的影响

软件产品属性	计算机系统属性	产品质量特性	主要用户使用质量	任务维护方使用质量	其他利益相关方关注的信息系统质量
➥	➥	功能性	✱		
➥	➥	性能效率	✱		✱
➥	➥	兼容性		✱	
➥	➥	易用性	✱		
➥	➥	可靠性	✱		✱
➥	➥	信息安全性	✱		✱
➥	➥	维护性		✱	
➥	➥	可移植性		✱	

注：➥ 这些属性影响产品质量。
✱ 产品质量影响这些利益相关方的使用质量。

附 录 A
（资料性附录）
与 GB/T 16260.1—2006 中质量模型的对比

本部分修订了 GB/T 16260.1—2006 中的质量模型，其主要技术差异如下：

a） GB/T 16260.1—2006 中的使用质量模型提供了 4 个特性，没有子特性。本部分将使用质量模型的特性调整为 5 个，并增加了子特性，修订细节如下：
   1） 将特性生产率更名为效率；
   2） 删除了安全性特性；
   3） 增加了抗风险及其子特性（经济风险缓解性、健康和安全风险缓解性、环境风险缓解性）和周境覆盖及其子特性（周境完备性、灵活性）；
   4） 增加了满意度的子特性，包括有用性、可信性、愉悦性和舒适性。

b） 本部分将 GB/T 16260.1—2006 中的外部和内部质量模型更名为产品质量模型，并将其 6 个特性调整为 8 个特性，修订细节如下：
   1） 将功能性的安全保密性子特性提升为一个独立的特性，并更名为信息安全性，其子特性包括保密性、完整性、抗抵赖性、可核查性、真实性和信息安全性的依从性；
   2） 增加了兼容性特性，其子特性包括共存性（它原本是可移植性特性的子特性）、互操作性（它原本是功能性特性的子特性）和兼容性的依从性；
   3） 还增加了如下子特性：功能性的功能完备性子特性、性能效率的容量子特性、易用性的用户差错防御性子特性和易访问性子特性、可靠性的可用性子特性、维护性的模块化子特性和可重用性子特性；
   4） 将维护性的易改变性和稳定性合并成新的子特性，即易修改性；
   5） 修改了特性或子特性的名称，新名称更准确，包括：准确性更名为功能正确性、适合性更名为功能适合性、效率更名为性能效率、效率依从性更名为性能效率的依从性、易理解性更名为可辨识性、吸引性更名为用户界面舒适性。

c） 在合适的情况下，对特性及子特性的阐述宜扩展到计算机系统的范畴，而不是仅限于软件范畴。

表 A.1 列出了本部分中质量特性和子特性与 GB/T 16260.1—2006 中的对比结果。

表 A.1 本部分与 GB/T 16260.1—2006 中质量模型的对比

条款	GB/T 25000.10	GB/T 16260.1	注释
4.2.1	使用质量模型	使用质量	使用质量现在指的是一种系统质量
4.2.2.1	有效性	有效性	
4.2.2.2	效率	生产率	名字与 ISO/IEC 25062 和 ISO/IEC 9241-11 相对应
4.2.2.3	满意度	满意度	
4.2.2.3.1	有用性		新增的子特性
4.2.2.3.2	可信性		新增的子特性
4.2.2.3.3	愉悦度		新增的子特性
4.2.2.3.4	舒适性		新增的子特性

表 A.1（续）

条款	GB/T 25000.10	GB/T 16260.1	注释
4.2.2.4	抗风险	安全性	新增的特性,同时删除了安全性特性
4.2.2.4.1	经济风险缓解性		新增的子特性
4.2.2.4.2	健康和安全风险缓解性		新增的子特性
4.2.2.4.3	环境风险缓解性		新增的子特性
4.2.2.5	周境覆盖		新增的特性,明确了隐含的质量问题
4.2.2.5.1	周境完备性		新增的子特性(产品在所有必需的使用周境中是可用的,这点至关重要)
4.2.2.5.2	灵活性		新增的子特性(确保产品被用在新的使用周境中)
4.3.1	产品质量模型	内部和外部质量	内部和外部质量统称为产品质量
4.3.2.1	功能性	功能性	
4.3.2.1.1	功能完备性		新增的子特性,覆盖了明确的要求
4.3.2.1.2	功能正确性	准确性	比准确性更通用
4.3.2.1.3	功能适合性	适合性	新的名字更加准确,覆盖了隐含的要求
		互操作性	移至兼容性子特性中
		安全保密性	提升为新的特性,并更名为信息安全性
4.3.2.1.4	功能性的依从性	功能性的依从性	
4.3.2.2	性能效率	效率	重命名是为了避免与 ISO/IEC 25062 中效率的定义互相冲突
4.3.2.2.1	时间特性	时间特性	
4.3.2.2.2	资源利用性	资源利用性	
4.3.2.2.3	容量		新增的子特性(尤其适用于计算机系统)
4.3.2.2.4	性能效率的依从性	效率依从性	新的名字更加准确
4.3.2.3	兼容性		新增的特性
4.3.2.3.1	共存性	共存性	源于可移植性的子特性
4.3.2.3.2	互操作性		源于功能性的子特性
4.3.2.3.3	兼容性的依从性		新增的子特性
4.3.2.4	易用性		明确了隐含的质量问题
4.3.2.4.1	可辨识性	易理解性	新的名字更加准确
4.3.2.4.2	易学性	易学性	
4.3.2.4.3	易操作性	易操作性	
4.3.2.4.4	用户差错防御性		新增的特性(对达到抗风险目标尤其重要)
4.3.2.4.5	用户界面舒适性	吸引性	新的名字更加准确
4.3.2.4.6	易访问性		新增的子特性
4.3.2.4.7	易用性的依从性	易用性的依从性	

表 A.1（续）

条款	GB/T 25000.10	GB/T 16260.1	注释
4.3.2.5	可靠性	可靠性	
4.3.2.5.1	成熟性	成熟性	
4.3.2.5.2	可用性		新增的子特性
4.3.2.5.3	容错性	容错性	
4.3.2.5.4	易恢复性	易恢复性	
4.3.2.5.5	可靠性的依从性	可靠性的依从性	
4.3.2.6	信息安全性	安全保密性	将功能性的安全保密性子特性提升为一个独立的特性，并更名为信息安全性
4.3.2.6.1	保密性		新增的子特性
4.3.2.6.2	完整性		新增的子特性
4.3.2.6.3	抗抵赖性		新增的子特性
4.3.2.6.4	可核查性		新增的子特性
4.3.2.6.5	真实性		新增的子特性
4.3.2.6.6	信息安全性的依从性		新增的子特性
4.3.2.7	维护性	维护性	
4.3.2.7.1	模块化		新增的子特性
4.3.2.7.2	可重用性		新增的子特性
4.3.2.7.3	易分析性	易分析性	
4.3.2.7.4	易修改性	易改变性	将易改变性和稳定性合并为易修改性，新的名字更加准确
		稳定性	
4.3.2.7.5	易测试性	易测试性	
4.3.2.7.6	维护性的依从性	维护性的依从性	
4.3.2.8	可移植性	可移植性	
4.3.2.8.1	适应性	适应性	
4.3.2.8.2	易安装性	易安装性	
		共存性	移至兼容性的子特性中
4.3.2.8.3	易替换性	易替换性	
4.3.2.8.4	可移植性的依从性	可移植性的依从性	
注：ISO/IEC 25062 将转化为 GB/T 25000.62。			

附　录　B
（资料性附录）
与 ISO/IEC 25010:2011 的差异

本部分修改采用 ISO/IEC 25010:2011《系统与软件工程　系统与软件质量要求和评价(SQuaRE)　系统与软件质量模型》，与 ISO/IEC 25010:2011 相比，主要差异如下：

a) 修改了标准的名称。
b) 修改了引言中陈述的内容。
c) 删除了“2 符合性”。
d) 增加了规范性引用文件 ISO/IEC 25012:2008 和 ISO/IEC 25030:2007。
e) 整合了“4.3 一般术语”和“4.4 来自 ISO/IEC 25000 的术语”。为避免歧义，只采用其中关于“用户”的两个定义中的一个，即采用了“4.3.16 用户”，同时删除了“4.4.12 用户”。
f) 本部分的产品质量模型的每个特性中，均增加了依从性子特性，分别是：功能性的依从性、性能效率的依从性、兼容性的依从性、易用性的依从性、可靠性的依从性、信息安全性的依从性、维护性的依从性和可移植性的依从性。相应地，删除了 3.3 的注以及附录 A 中列项的第 5 项。
g) 附录 A 修改为本部分“与 GB/T 16260.1—2006 中质量模型的对比”。
h) 删除了 ISO/IEC 25010:2011 附录 B 的内容及其相关内容。
i) 本部分与 ISO/IEC 25010:2011 相比在结构上有所调整，详见表 B.1。

表 B.1　结构调整说明

ISO/IEC 25010:2011 章条编号	调整说明
2 Conformance	删除该条款，并新增“2 规范性引用文件”
3 Quality model framework	变更为“4 质量模型框架”
3.1 Quality models	提升至“4.1 结构”
3.2 Quality in use model	变更为“4.2.1 使用质量模型”，与 4.2.2 构成“4.2 使用质量模型及特性说明”
3.3 Product quality model	变更为“4.3.1 产品质量模型”，与 4.3.2 构成“4.3 产品质量模型及特性说明”
3.4 Targets of the quality models	变更为“4.4 质量模型的目标”
3.5 Using a quality model	变更为“4.5 质量模型的使用”
3.6 Quality from different stakeholder perspectives	变更为“4.6 来自不同利益相关方视角的质量”
3.7 Relationship between the models	变更为“4.7 各质量模型间的关系”
4.1 Quality in use model	变更为“4.2.2 特性说明”，与 4.2.1 构成“4.2 使用质量模型及特性说明”
4.2 Product quality model	变更为“4.3.2 特性说明”，与 4.3.1 构成“4.3 产品质量模型及特性说明”
4.3 General 和 4.4 Terms and definitions from ISO/IEC 25000	合并为“3 术语和定义”
Annex A(informative) Comparison with the quality model in ISO/IEC 9126-1	修改为“附录 A(资料性附录)与 GB/T 16260.1—2006 中质量模型的对比”
Annex B(informative) Example of mapping to dependability	删除该条款，并新增“附录 B(资料性附录)与 ISO/IEC 25010:2011 的差异”

附　录　C
（资料性附录）
利用质量模型测量

C.1　概述

本附录中的信息可能迁移到 GB/T 25000《系统与软件工程　系统与软件质量要求和评价(SQuaRE)》系列标准的其他部分的未来版本中。

C.2　软件质量测量模型

质量属性是软件所固有的性质，它有助于质量。质量属性可以归类为一个或多个(子)特性。

质量属性通过测量方法来测量。测量方法是一种逻辑操作序列，用来量化关于特定标度的属性。测量方法的结果被称为质量测度元素。质量特性和子特性可以通过测量函数来量化。测量函数是一个用来组合质量测度元素的算法。应用测量函数的结果被称为软件质量测度。这样说来，软件质量测度就是对质量特性和子特性的量化。多个软件质量测度可以被用来测量一个质量特性或子特性。

摘自 ISO/IEC 25020 的图 C.1 表明了本部分中的质量模型、ISO/IEC 2502n 中的测量和 ISO/IEC 15939 中提出的测量模型间的关系。

注：ISO/IEC 25020 将转化为 GB/T 25000.20。

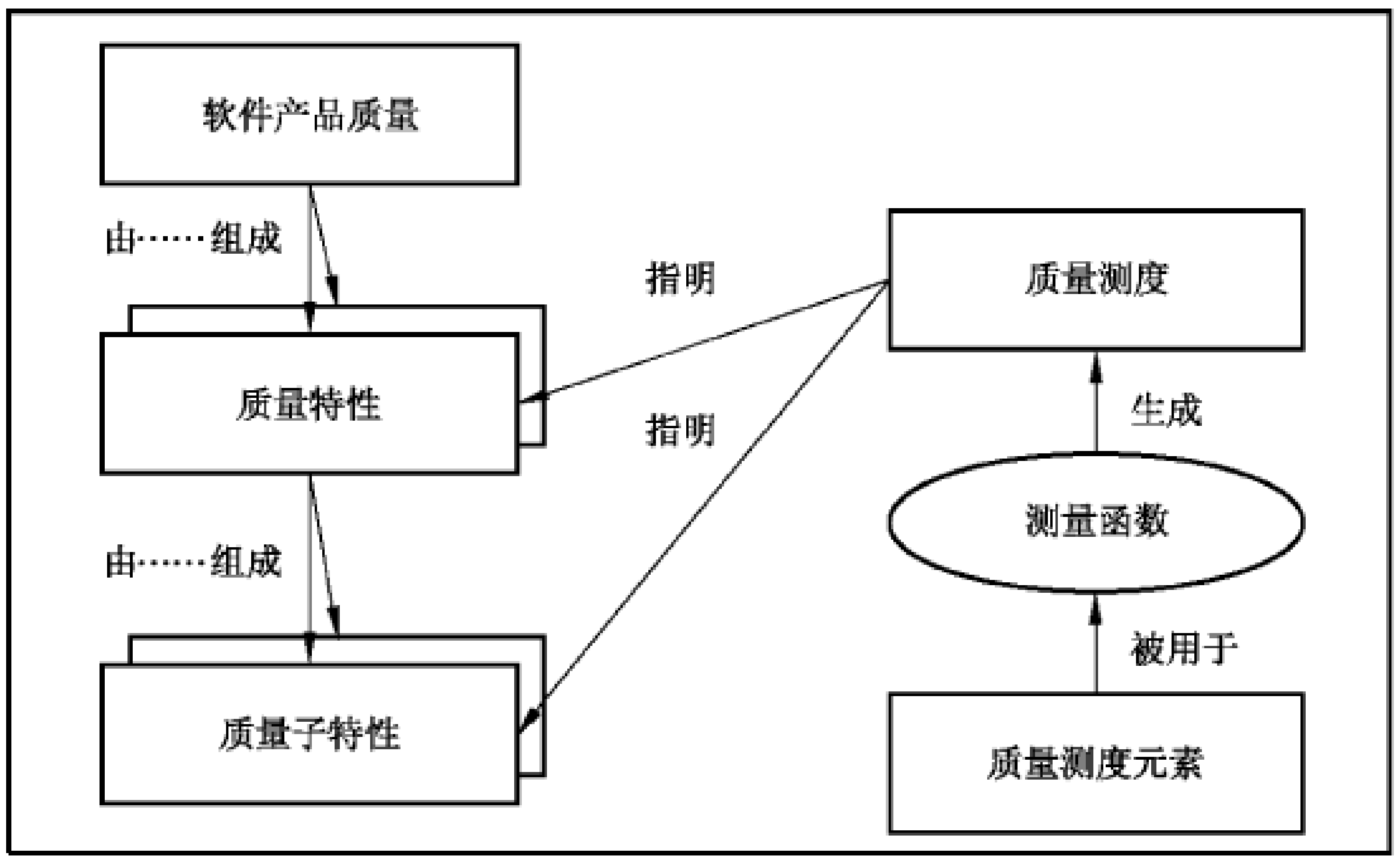


图 C.1　软件产品质量测量参考模型

C.3　质量途径

用户质量要求包括在指定的使用周境中对系统的使用质量的需求。当使用软件产品质量特性和子特性来规定质量的外部测度和内部测度时，这些已确定的要求可被利用。

软件产品质量可以通过测量内部属性(典型地是中间产品的静态测量)，或通过测量外部属性(典型地是测量代码执行时的行为)，或通过测量使用质量的属性(当产品在实际或模拟中使用时)来评价。(见图 C.2)

改进过程质量(在 ISO/IEC 12207 和 ISO/IEC 15288 中定义的任一生存周期过程的质量)有助于改进产品质量,而产品质量有助于改进系统使用质量。因此,评估和改进过程是改进产品质量的一种手段,而评价和改进产品质量则是改进系统使用质量的一种手段。同样,评价系统使用质量可以为改进产品提供反馈,而评价产品可以为改进过程提供反馈。

软件的适当的内部属性是达到要求的外部行为的先决条件,而适当的外部行为则是达到使用质量的先决条件(见图 C.2)。

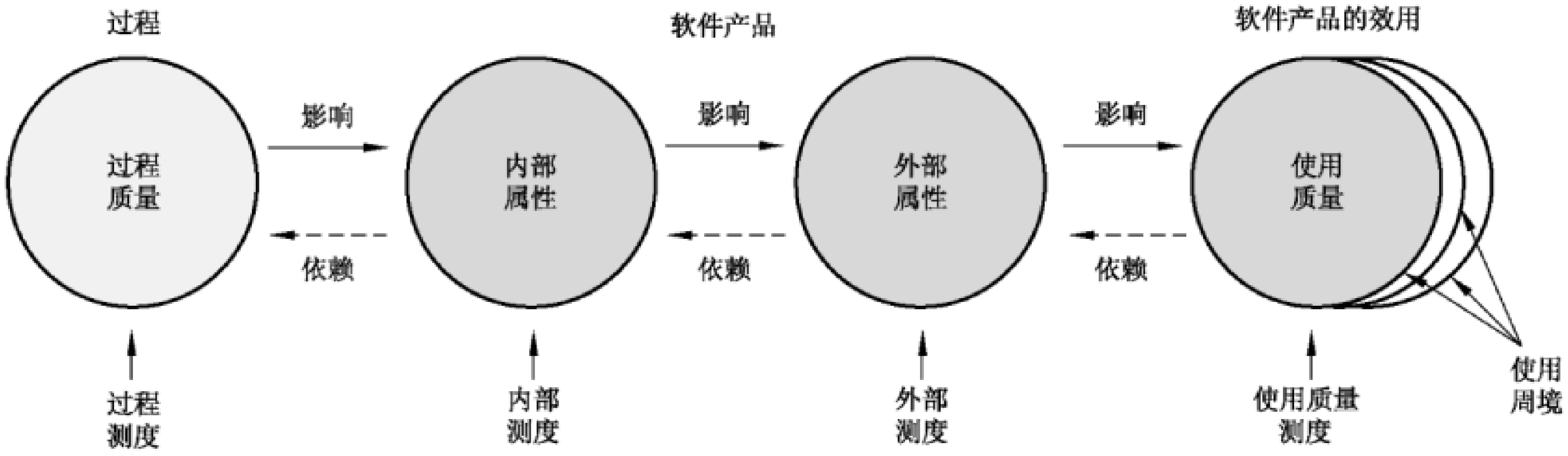


图 C.2　生存周期中的质量

C.4　质量影响

图 C.3 说明了质量模型的目标实体之间的关系。软件生存周期过程(如质量需求过程、设计过程和测试过程)影响着软件产品和系统质量。资源质量(如人力资源和用于生存周期过程的工具及技术)影响着过程质量,进而影响着产品质量。

不仅软件产品质量,还有系统的其他组件质量也影响着系统质量。质量影响(效用)是多方面的,这取决于使用周境。使用周境可以被定义为一个由用户、任务和环境组成的集合。表 1 中给出了使用周境的一些示例(见 4.6)。

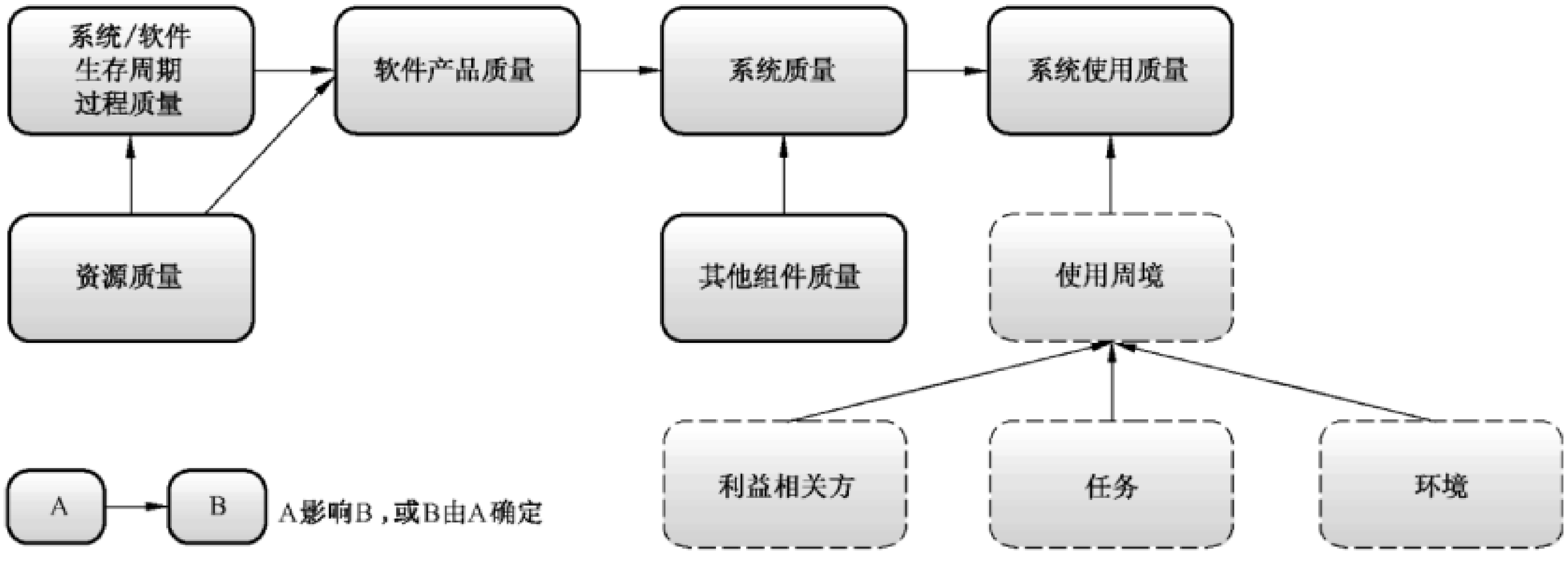


图 C.3　质量模型的目标实体及其相互间的关系

C.5　质量生存周期模型

ISO/IEC 25030 用图形解释了质量需求过程(见图 C.4)。图中的“利益相关方的要求”汇集了使用质量要求和产品质量要求,然后转化为特定的质量需求(利益相关方的需求)。

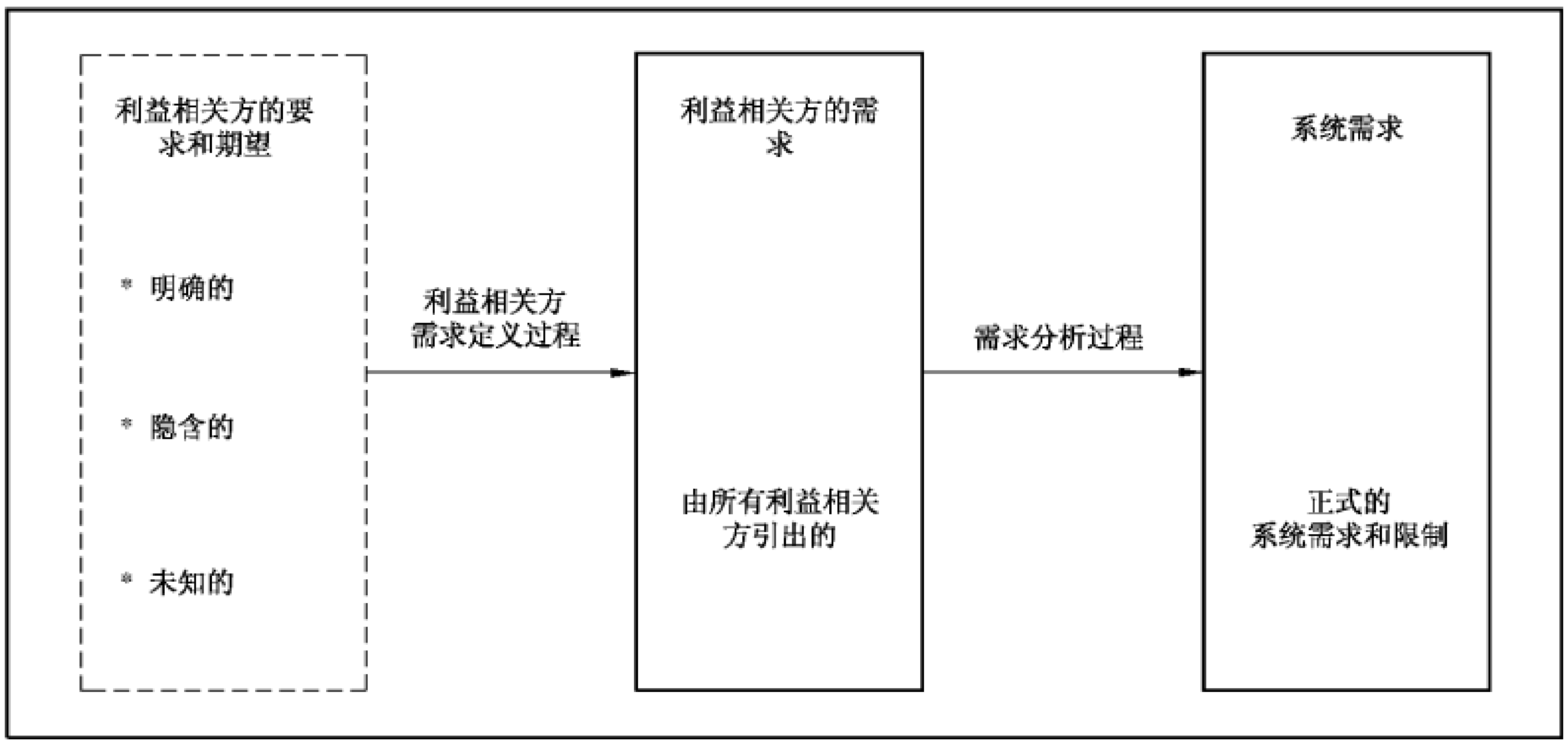


图 C.4　利益相关方需求定义和分析

质量生存周期模型(见图 C.5)涉及软件产品 3 个主要阶段中的质量:

a)　产品开发阶段是软件质量内部测度的主题;

b)　产品测试阶段是软件质量外部测度的主题;

c)　产品使用阶段是使用质量的主题。

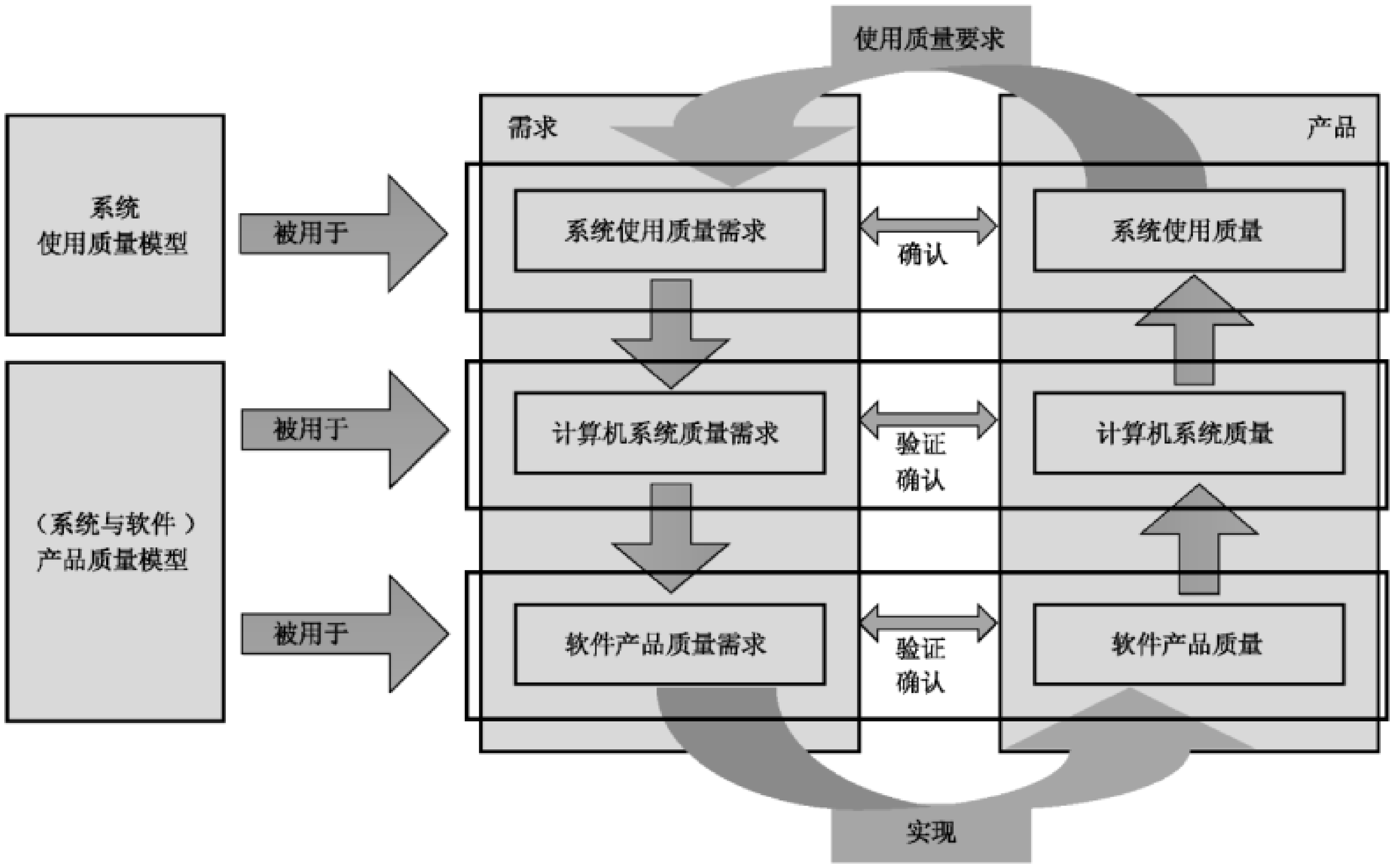


图 C.5　系统/软件质量生存周期模型

系统/软件质量生存周期模型还表明,对每种类型的质量而言,达到可接受的质量水平应该是其开发过程中不可或缺的一部分,包括:需求、实现和结果确认。

使用质量需求从用户视角规定要求的质量水平。这些需求是从用户和其他利益相关方(如软件开发方、系统集成方、需方或所有者)的要求中派生出来的。使用质量需求被用户用作软件产品确认的目标。对使用质量特性的需求,宜在质量需求规格说明中用使用质量测度准则加以明确,该准则可在评价产品时使用。

注 1：系统使用质量需求有助于标识和定义软件外部质量需求。

示例 1：指定类型的用户能够在指定时间内完成指定任务。

计算机系统质量的外部测度需求从外部视角规定要求的质量水平。它们包括来自利益相关方的质量需求(包括使用质量需求)。外部的软件质量需求被用作技术验证和软件产品确认的目标。对质量的外部测度需求,宜在质量需求规格说明中用外部测度准则量化地明确,该准则可在评价产品时使用。

注 2：质量的外部测度需求有助于标识和定义软件质量内部测度需求。

注 3：外部质量评价能用于预估系统使用质量。

示例 2：用户妥善应对错误信息并成功消除错误。

软件质量内部测度需求从产品的内部视角规定要求的质量水平。它们包括从外部质量需求派生的需求。软件质量内部测度需求被用来规定软件中间产品(规格说明、源代码等)的性质。软件内部质量需求也可以被用来规定可交付的、非可执行的软件产品(如文档和手册)的性质。软件质量内部测度需求可以被用作各个开发阶段的验证目标,还可以用来定义开发策略和开发期间的评价与验证准则。

注 4：软件质量内部测度能用于预估软件质量外部测度。

示例 3：对所有错误信息规定其纠正措施,并且用户输入均可被撤消。

ISO/IEC 25030 描述了软件质量需求,ISO/IEC 25040 描述了软件质量评价过程。

在开发期间,质量模型与相关的测度可以用于设计和实施的管理,以确保质量目标得到满足。质量模型与相关测度的一个重要应用是能够较早地洞察软件质量情况。这种洞察力可以用来管理整个生命周期中的质量和预估在使用中质量目标是否有可能得到满足。

注 5：在有合同的环境或规范的环境中,例如核安全领域,要求是特定的;而在其他的环境中,则应对隐含的要求进行标识和定义(ISO 8402:1994,2.1,注 1)。

C.6 软件性质

一些软件性质是软件产品固有的,一些是被赋予给产品的。而在一个特定使用周境下,软件产品质量是由其固有性质决定的。

注 1：例如,固有性质是指代码的行数和软件提供的计算数值的精度;赋予的性质是指一个软件产品的所有者、保修单和软件价格。

固有性质可以被分类为功能性性质或者质量性质。功能性性质决定了软件能做什么;质量性质决定了软件执行的有多好。质量性质是软件与相关系统所固有的。因此被赋予的性质不属于软件的质量特性范畴,因为它可以被更改而无需更改软件。图 C.6 给出了软件性质的分类。

软件性质	固有性质	特定领域的功能性性质
		质量性质(如功能性、可靠性、性能效率、易用性、信息安全性、兼容性、维护性、可移植性)
	赋予的性质	管理性质(如价格、交货日期、产品前景、产品供应商)

图 C.6 软件性质

注 2：特定领域的功能性性质通常只关注输入数据转换为输出数据。为了实现质量性质和约束,需要更多的功能。

C.7 内部测度、外部测度和使用质量测度

对每个子特性而言,软件能力是由一组可测量的静态的内部性质决定的。GB/T 16260.3(将被 GB/T 25000.23 取代)给出了内部测度的实例。根据系统(包含软件)提供能力的程度,可从外部来测量特性及子特性。GB/T 16260.2(将被 GB/T 25000.23 取代)给出了外部测度的实例。

注 1：在 ISO/IEC 25023 中不再区分内部测度和外部测度。

注 2：ISO/IEC 25023 将转化为 GB/T 25000.23。

系统/软件质量外部测度提供了系统/软件的“黑盒”观点，并处理与计算机硬件及操作系统上的软件执行有关的性质。软件质量内部测度提供了软件的“白盒”观点，并处理软件产品静态性质，这通常在开发期间可被评价。从内部测量的软件质量会对从外部测量的系统/软件质量有影响，它又会对系统使用质量有影响。

示例：根据 ISO 9241-14 中菜单界面设计指南的一致性程度从内部测出的易操作性，将有助于根据菜单操作的成功程度从外部测量易操作性，这将有助于执行任务时的有效性、效率和满意度(使用质量)。

基于检测静态性质的内部测度可以用来测量软件工作产品的固有性质(见表 C.1)。静态分析方法包括检测和自动分析工具。工作产品包括需求和设计文档、源代码和测试流程。

动态性质的外部测度可用来测量计算机系统的固有性质(见图 5 中的目标计算机系统)，以及软件产品的系统依赖性。

使用质量测度(从测试或观察真实或模拟使用的结果中派生而来)用来测量系统的固有性质，该系统可包含软件、硬件、通信和用户，以及软件密集型计算机系统或软件产品的系统依赖性。使用质量测度关系到系统对其利益相关方的影响。

表 C.1　内部质量测度、外部质量测度和使用质量测度的差异

测量的性质类型	软件产品性质	计算机系统行为性质	人机系统影响性质
质量测度类型	内部的：静态性质检测	外部的：动态性质的测试和建模	使用质量：对真实或模拟使用的结果的测试或观察
软件产品性质类型	固有的	计算机系统依赖的	人机系统依赖的
计算机系统性质类型		固有的	人机系统依赖的
人机系统性质类型			固有的

软件内部测度可在系统/软件开发过程的早期阶段用来预测系统/软件质量外部测度。对同一性质，通常存在内部测度和外部测度，例如用于估计预期响应时间的内部测度，可用来预测从外部测得的时间。

C.8　产品质量和数据质量的关系

ISO/IEC 25012：2008 中的数据质量模型是产品质量模型的补充：

a)　固有的数据质量(见表 C.2)和软件质量内部测度(见表 C.1)都有助于计算机系统的总体质量。

b)　系统依赖的数据质量测度和软件质量外部测度在计算机系统方面的评估类似。其区别在于，系统依赖的数据质量测度关注于数据对计算机系统质量做出的贡献，而软件质量外部测度关注于软件的贡献。但二者实际上测量的都是计算机系统的性质。

表 C.2　数据固有性质与计算机系统性质间的关系

测量的性质类型	数据固有性质	计算机系统性质
质量测度类型	固有的数据质量	系统依赖的数据质量
数据性质类型	固有的	计算机系统依赖的
计算机系统性质类型		固有的

参 考 文 献

[1] GB/T 11457—2006 信息技术 软件工程术语

[2] GB/T 16260.1—2006 软件工程 产品质量 第1部分:质量模型(ISO/IEC 9126-1:2001,IDT)

[3] GB/T 16260.2—2006 软件工程 产品质量 第2部分:外部度量(ISO/IEC TR 9126-2:2003,IDT)

[4] GB/T 16260.3—2006 软件工程 产品质量 第3部分:内部度量(ISO/IEC TR 9126-3:2003,IDT)

[5] GB/T 16260.4—2006 软件工程 产品质量 第4部分:使用质量的度量(ISO/IEC TR 9126-4:2004,IDT)

[6] GB/T 18492—2001 信息技术 系统及软件完整性级别(ISO/IEC 15026:1998,IDT)

[7] GB/T 18978.11—2004 使用视觉显示终端(VDTs)办公的人类工效学要求 第11部分:可用性指南(ISO 9241-11:1998,IDT)

[8] GB/T 19001—2008 质量管理体系 要求(ISO 9001:2008,IDT)

[9] GB/T 19715.1—2005 信息技术 信息技术安全管理指南 第1部分:信息技术安全概念和模型(ISO/IEC TR 13335-1:1996,IDT)

[10] GB/T 25000.1—2010 软件工程 软件产品质量要求和评价(SQuaRE) SQuaRE指南(ISO/IEC 25000:2005,IDT)

[11] IEC 60050-191 Ed.2.0 International Electrotechnical Vocabulary—Part 191:Dependability

[12] ISO 8402:1994 Quality management and quality assurance—Vocabulary

[13] ISO 9241-14:1997 Ergonomic requirements for office work with visual displa terminals (VDTs)—Part 14:Menu dialogues

[14] ISO 9241-110:2006 Ergonomics of human-system interaction—Part 110:Dialogue principles

[15] ISO/IEC 12207:2008 Systems and software engineering—Software life cycle processes

[16] ISO/IEC 14598-2:2002 Software engineering—Product evaluation—Part 2:Planning and management

[17] ISO/IEC 14598-3:2002 Software engineering—Product evaluation—Part 3:Process for developers

[18] ISO/IEC 14598-4:1999 Software engineering—Product evaluation—Part 4:Process for acquirers

[19] ISO/IEC 14598-5:1998 Information technology—Software product evaluation—Part 5:Process for evaluators

[20] ISO/IEC 14598-6:2001 Software engineering—Product evaluation—Part 6: Documentation of evaluation modules

[21] ISO/IEC 15288:2008 Systems and software engineering—Software life cycle processes

[22] ISO/IEC 15504(parts 1 to 5) Information technology—Process assessment

[23] ISO/IEC 15939:2007 Systems and Software engineering—Measuement process

[24] ISO/IEC 25020:2007 Software Engineering—Software product Quality Rquirements and Evaluation(SQuaRE)—Measurement reference model and guide

[25] ISO/IEC 25021:2012 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Quality measure elements

[26] ISO/IEC DIS 25022.2 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Measurement of quality in use

[27] ISO/IEC DIS 25023 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Measurement of systems and software product quality

[28] ISO/IEC 25040:2011 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Evaluation process

[29] ISO/IEC 25062:2006 Software engineering—Software product Quality Requirements and Evaluation (SQuaRE)—Common Industry Format (CIF) for usability test reports

[30] ISO/IEC/IEEE 24765:2010 Systems and software engineering—Vocabulary

GB/T 25000.10—2016

中 华 人 民 共 和 国

国 家 标 准

系统与软件工程

系统与软件质量要求和评价(SQuaRE)

第 10 部分:系统与软件质量模型

GB/T 25000.10—2016

*

中 国 标 准 出 版 社 出 版 发 行

北京市朝阳区和平里西街甲 2 号(100029)

北京市西城区三里河北街 16 号(100045)

网址:www.spc.org.cn

服务热线:400-168-0010

2016 年 11 月第一版

*

书号:155066 · 1-55374

版权专有 侵权必究

GB/T 25000.10-2016